

In the Claims

1. (Currently amended) A process for reordering items ~~retrieved from in~~ a database to be retrieved for display to a user, comprising the steps of:

accepting user input from a keyboard;

providing a linguistic database (LDB);

~~wherein said linguistic database contains~~ words ordered using according to a linguistic frequency of use model;

providing a user database (UDB) separate from the LDB which stores user defined words that the user specifically enters into the system and which includes a reorder database (RDB) that stores database object numbers;

dynamically retrieving words from said linguistic database LDB and from said UDB words that comprise include the sequence of letters formed by the user's keyboard input;

displaying to the user a list of said retrieved words to the user;

~~wherein words in said list are ordered using the linguistic database ordering and a overridden by any existing dynamic reordering frequency count;~~

~~wherein the dynamic reordering frequency count overrides the linguistic database ordering for words in said list;~~

enabling the user to select a word from the displayed list; and

assigning a dynamic reordering frequency count to words selected by the user from said list and inserting the selected words' assigned reordering frequency counts and object numbers into said reorder database.

2. (Canceled)

3. (Canceled)

4. (Currently amended) The process of Claim 21, wherein said assigning step inserts into said reorder database a first ordered word from said list and a non first ordered word from said list into said reorder database if the user has selected the non

first ordered word for the first time, and wherein ~~the~~ inserts into said reorder database the nonselected first ordered word ~~is inserted~~ if it does not already exist in said reorder database.

5. (Currently amended) The process of Claim 4, wherein ~~the first ordered word in said list loses its position~~ if the non first ordered word is selected by the user a predetermined number of times, ~~and wherein~~ then the non first ordered word is ~~then~~ assigned a higher frequency value count than the first ordered word.
6. (Currently amended) The process of Claim 4, wherein all non first ordered words entered into said reorder database are initially assigned equal reordering frequencies counts.
7. (Currently amended) The process of Claim 1, wherein a word's reordering frequency count is increased each time the user selects the word.
8. (Currently amended) The process of Claim 1, wherein if the user selects from said list a word ~~in said list is selected by the user and the word which~~ is below a second ordered position then said assigning step assigns a value to the word's reordering frequency count ~~to a value that places the word in the second ordered position in said list~~.
9. (Currently amended) The process of Claim 21, further comprising the step of: periodically ~~performing an aging process to the reordering frequencies~~ counts in said reorder database; and ~~wherein by reducing the reordering frequency counts in said reordering database are reduced by a predetermined factor by said aging step.~~
10. (Currently amended) The process of Claim 31, further comprising the step of: periodically checking ~~the free space of~~ in said reordering database;

~~_____ wherein and, if the free space in said reordering database below is less than a predetermined threshold amount, then removing from said reorder database words that have reordering frequency counts below a predetermined threshold from said reordering database.~~

11. (Currently amended) The process of Claim 10, wherein said ~~checking removing~~ step removes user defined words having reordering frequency counts below the predetermined threshold after removing other words having reordering frequency counts below the predetermined threshold ~~from said reordering database.~~

12. (Currently amended) The process of Claim 1, further comprising the step of: resolving reordering frequency collisions ~~in said list;~~

~~_____ wherein said resolving step resolves a collision if when two words in said list have the same equal reordering frequency counts by ordering the most recently selected of the two words first.~~

13. (Currently amended) The process of Claim 1, further comprising the step of: resolving reordering frequency collisions ~~in said list;~~

~~_____ wherein said resolving step resolves a collision if when two words in said list have the same equal reordering frequency counts by ordering the word having a higher initial ordering in said linguistic database first.~~

14. (Currently amended) The process of Claim 21, further comprising the step of: resolving reordering frequency collisions in said list;

~~_____ wherein said resolving step resolves a collision if two words have the same reordering frequency by ordering user defined words first if when a user defined word and a word from said linguistic database have a collision equal reordering frequency counts by ordering the user defined word first.~~

15. (Original) The process of Claim 1, wherein words selected by the user that do not have a possibility of a collision with other words are not assigned a reordering frequency count.

16. (Currently amended) An apparatus for reordering items ~~retrieved from in a~~ database to be retrieved for display to a user, comprising:

a module for accepting user input from a keyboard;

a linguistic database (LDB);

~~wherein said linguistic database contains~~ing words ordered using according to a linguistic frequency of use model;

providing a user database (UDB) separate from the LDB which stores user defined words that the user specifically enters into the system and which includes a reorder database (RDB) that stores database object numbers;

a module for dynamically retrieving words ~~from said linguistic database LDB and from said UDB words~~ that comprise include the sequence of letters formed by the user's keyboard input;

a module for displaying to the user a list of said retrieved words ~~to the user;~~

~~wherein words in said list are ordered using the linguistic database ordering and a~~ overridden by any existing dynamic reordering frequency count;

~~wherein the dynamic reordering frequency count overrides the linguistic database ordering for words in said list;~~

enabling the user to select a word from the displayed list; and

a module for assigning a dynamic reordering frequency count to words selected by the user ~~from said list~~ and inserting the selected words' assigned reordering frequency counts and object numbers into said reorder database.

17. (Canceled)

18. (Canceled)

19. (Currently amended) The apparatus of Claim ~~47~~16, wherein said assigning module inserts into said reorder database a first ordered word from said list and a non first ordered word from said list into said reorder database if the user has selecteds the non first ordered word for the first time, and ~~wherein the~~ inserts into said reorder database the nonselected first ordered word ~~is inserted~~ if it does not already exist in said reorder database.

20. (Currently amended) The apparatus of Claim 19, wherein ~~the first ordered word in said list loses its position~~ if the non first ordered word is selected by the user a predetermined number of times, ~~and wherein then~~ the non first ordered word is then assigned a higher frequency value count than the first ordered word.

21. (Currently amended) The apparatus of Claim 19, wherein all non first ordered words entered into said reorder database are initially assigned equal reordering frequencies counts.

22. (Currently amended) The apparatus of Claim 16, wherein a word's reordering frequency count is increased each time the user selects the word.

23. (Currently amended) The apparatus of Claim 16, wherein if the user selects from said list a word ~~in said list is selected by the user and the word which~~ is below a second ordered position then said assigning module assigns a value to the word's reordering frequency count ~~to a value~~ that places the word in the second ordered position in said list.

24. (Currently amended) The apparatus of Claim ~~47~~16, further comprising:
a module for periodically ~~performing an aging process to the reordering frequencies~~ counts in said reorder database; and
~~wherein by reducing the reordering frequency counts in said reordering database are reduced by a predetermined factor by said aging module.~~

25. (Currently amended) The apparatus of Claim 16, further comprising:
a module for periodically checking the free space ~~in~~ said reordering database;
~~wherein and, if the free space in said reordering database below is less than a~~
predetermined ~~threshold amount~~, then removing ~~from said reorder database~~ words that
have reordering frequency counts below a predetermined threshold ~~from said~~
reordering database.
26. (Currently amended) The apparatus of Claim 25, wherein said ~~checking~~
removing module removes user defined words having reordering frequency counts
below the predetermined threshold after removing other words having reordering
frequency counts below the predetermined threshold ~~from said reordering database~~.
27. (Currently amended) The apparatus of Claim 16, further comprising:
a module for resolving reordering frequency collisions ~~in said list~~;
~~wherein said resolving module resolves a collision if when two words in said list~~
have the ~~same~~ equal reordering frequency counts by ordering the most recently selected
of the two words first.
28. (Currently amended) The apparatus of Claim 16, further comprising:
a module for resolving reordering frequency collisions ~~in said list~~;
~~wherein said resolving module resolves a collision if when two words in said list~~
have the ~~same~~ equal reordering frequency counts by ordering the word having a higher
initial ordering in said linguistic database first.
29. (Currently amended) The apparatus of Claim 16, further comprising:
a module for resolving reordering frequency collisions in said list;
~~wherein said resolving module resolves a collision if two words have the same~~
~~reordering frequency by ordering user defined words first if when a user defined word~~
and a word from said linguistic database have a collision equal reordering frequency
counts by ordering the user defined word first.

30. (Original) The apparatus of Claim 16, wherein words selected by the user that do not have a possibility of a collision with other words are not assigned a reordering frequency count.

31. (Currently amended) ~~A~~The process of Claim 1 for reordering items retrieved from a database for display to a user, comprising the steps of:

- ~~— accepting user input from a keyboard;~~
- ~~— providing a linguistic database;~~
- ~~— wherein said linguistic database contains words ordered using a linguistic model and an ordering frequency for each word;~~
- ~~— dynamically retrieving words from said linguistic database that comprise letters formed by the user's keyboard input;~~
- ~~— displaying a list of said retrieved words to the user;~~
- ~~— wherein words in said list are ordered using the linguistic database ordering and a dynamic reordering frequency count;~~
- ~~— wherein the dynamic reordering frequency count overrides the linguistic database ordering for words in said list; and~~
- ~~— wherein when a word from the LDB is selected for the first time, said step of assigning a dynamic reordering frequency count to words selected by the user from said list and uses the word's frequency of use order in the LDB as an initializing the assigned dynamic reordering frequency count for a word to the ordering frequency value obtained from said linguistic database for the word.~~

32. (Canceled)

33. (Canceled)

34. (Currently amended) The process of Claim ~~33~~31, wherein all words specifically entered by the user are initially assigned equal reordering frequencies counts by said assigning step.

Claims 35-44 (Canceled)

45. (Currently amended) ~~An~~The apparatus of Claim 16 for reordering items retrieved from a database for display to a user, comprising:

- ~~— a module for accepting user input from a keyboard;~~
- ~~— a linguistic database;~~
- ~~— wherein said linguistic database contains words ordered using a linguistic model and an ordering frequency for each word;~~
- ~~— a module for dynamically retrieving words from said linguistic database that comprise letters formed by the user's keyboard input;~~
- ~~— a module for displaying a list of said retrieved words to the user;~~
- ~~— wherein words in said list are ordered using the linguistic database ordering and a dynamic reordering frequency count;~~
- ~~— wherein the dynamic reordering frequency count overrides the linguistic database ordering for words in said list; and~~
- ~~— a~~wherein when a word from the LDB is selected for the first time, said module for assigning a dynamic reordering frequency count to words selected by the user from said list and uses the word's frequency of use order in the LDB as an initializing the assigned dynamic reordering frequency count for a word to the ordering frequency value obtained from said linguistic database for the word.

46. (Canceled)

47. (Canceled)

48. (Currently amended) The apparatus of Claim 4745, wherein all words specifically entered by the user are initially assigned equal reordering frequenciesy counts by said assigning module.

Claims 49-58 (Canceled)

59. (New) A method of dynamically reordering a database to display words associated with key presses in an order based on a user's use of the words, comprising:
providing a fixed linguistic database of words ordered according to a linguistic frequency of use model;
providing a user database, separate from said linguistic database, for storing words selected by a user and storing frequency of use information for said selected words;
monitoring key presses and retrieving from said databases words which begin with characters corresponding to said key presses;
displaying a list of retrieved words in order of their frequencies of use as determined by the frequency of use model overridden by any existing frequency of use information;
enabling said user to select a desired word from the displayed list; and
updating the frequency of use information stored in the user database for the selected word.
60. (New) The method of Claim 59 wherein said user database stores each word as a key press sequence and a corresponding database object number.